List Statistical KPMG about Autonomous Vehicles

Manuel Robalinho

2018/09

Plotting KPMG information Autonomous Vehicle Readiness Index

https://assets.kpmg.com/content/dam/kpmg/tw/pdf/2018/03/KPMG-Autonomous-Vehicle-Readiness-Index.pdf (https://assets.kpmg.com/content/dam/kpmg/tw/pdf/2018/03/KPMG-Autonomous-Vehicle-Readiness-Index.pdf)

In [1]:

```
%matplotlib inline

import matplotlib.pyplot as plt
import matplotlib.cm as cm
import matplotlib.font_manager as fm
import matplotlib.patches as mpatches
import geopandas as gpd
import pandas as pd
import numpy as np
import seaborn as sns
```

In [2]:

```
path = 'ML/'
```

In [3]:

```
# Formating Graphics
title_font = fm.FontProperties(family='Arial', style='normal', size=20, weight='normal'
, stretch='normal')
label_font = fm.FontProperties(family='Arial', style='normal', size=16, weight='normal'
, stretch='normal')
ticks_font = fm.FontProperties(family='Arial', style='normal', size=12, weight='normal'
, stretch='normal')
annotation_font = fm.FontProperties(family='Arial', style='normal', size=11, weight='no
rmal', stretch='normal')
```

In [4]:

```
# plot the coordinates over a country-level map.
world = gpd.read_file(gpd.datasets.get_path('naturalearth_lowres'))
```

In [5]:

```
# https://matplotlib.org/examples/color/named_colors.html

# List of colors
color_list = ['red', 'tomato', 'lightsalmon', 'magenta','deepskyblue','palegreen',
              'plum','cyan', 'darkorange','wheat','seashell',
              'linen','aliceblue','azure','mintcream', 'oldlace',
              'floralwhite','honeydew','cornsilk','lightyellow','whitesmoke',
              'lavender','lightcyan','white','ghostwhite','snow',
              'oldlace','oldlace','oldlace','oldlace']
```

In [6]:

```
# Information KPMG
# Import file
df_kpmg = pd.read_excel(path + 'Assessing countries openness and preparedness for auton
omous vehicles.xls')
```

In [7]:

```
df_kpmg.head()
```

Out[7]:

	Overall rank	Country	Total score	Policy and legislation Rank	Policy and legislation Score	Technology & innovation Rank	Technology & innovation Score	Infrast
0	14	Australia	19.40	11	6.01	13	3.18	9
1	12	Austria	20.00	9	6.73	11	3.69	8
2	17	Brazil	7.17	20	0.93	18	0.86	19
3	7	Canada	22.61	7	7.12	6	4.97	11
4	16	China	13.94	16	4.38	15	2.25	15

```
In [8]:
```

```
# Transform table is a Pandas dataframe
df_kpmg = pd.DataFrame(df_kpmg)
df_kpmg.head()
```

```
Out[8]:
```

	Overall rank	Country	Total score	Policy and legislation Rank	Policy and legislation Score	Technology & innovation Rank	Technology & innovation Score	Infrastru
0	14	Australia	19.40	11	6.01	13	3.18	9
1	12	Austria	20.00	9	6.73	11	3.69	8
2	17	Brazil	7.17	20	0.93	18	0.86	19
3	7	Canada	22.61	7	7.12	6	4.97	11
4	16	China	13.94	16	4.38	15	2.25	15

Country Overall ranking

In [27]:

```
# Sort table by Overall rank
countdata = df_kpmg.sort_values(by=['Total score'],ascending=False)
countdata.head(20)
```

Out[27]:

	Overall rank	Country	Total score	Policy and legislation Rank	Policy and legislation Score	Technology & innovation Rank	Technology & innovation Score	Infra
11	1	Netherlands	27.73	3	7.89	4	5.46	1
14	2	Singapore	26.08	1	8.49	8	4.26	2
19	3	United States	24.75	10	6.38	1	6.97	7
16	4	Sweden	24.73	8	6.83	2	6.44	6
18	5	United Kingdom	23.99	4	7.55	5	5.28	10
6	6	Germany	22.74	5	7.33	3	6.15	12
3	7	Canada	22.61	7	7.12	6	4.97	11
17	8	United Arab Emirates	20.89	6	7.26	14	2.71	5
12	9	New Zealand	20.75	2	7.92	12	3.26	16
9	10	Korea	20.71	14	5.78	9	4.24	4
8	11	Japan	20.28	12	5.93	7	4.79	3
1	12	Austria	20.00	9	6.73	11	3.69	8
5	13	France	19.44	13	5.92	10	4.03	13
0	14	Australia	19.40	11	6.01	13	3.18	9
15	15	Spain	14.58	15	4.95	16	2.21	14
4	16	China	13.94	16	4.38	15	2.25	15
2	17	Brazil	7.17	20	0.93	18	0.86	19
13	18	Russia	7.09	17	2.58	20	0.52	20
10	19	Mexico	6.51	19	1.16	17	1.01	17
7	20	India	6.14	18	1.41	19	0.54	18

In [28]:

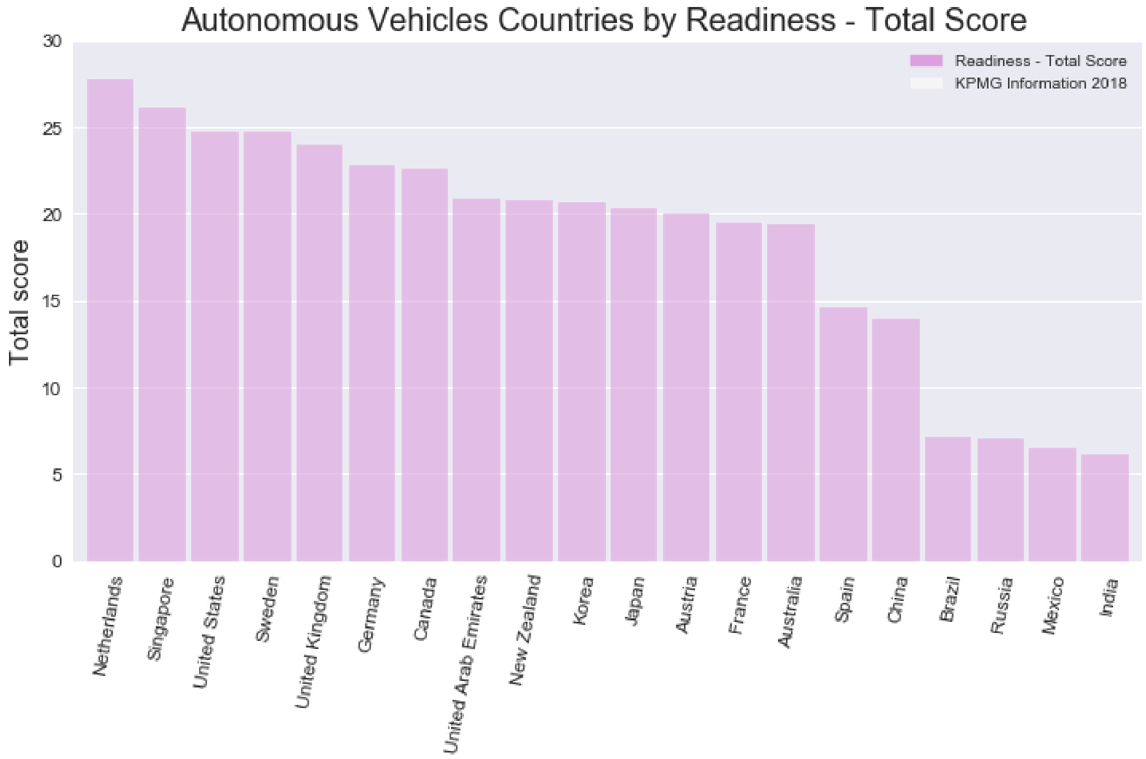
```
# Filter first 20 lines
countdata = countdata[countdata['Overall rank'] <= 20]

# make a copy with specified columns
data_plot = countdata[['Country', 'Total score']].copy()
data_plot
```

Out[28]:

	Country	Total score
11	Netherlands	27.73
14	Singapore	26.08
19	United States	24.75
16	Sweden	24.73
18	United Kingdom	23.99
6	Germany	22.74
3	Canada	22.61
17	United Arab Emirates	20.89
12	New Zealand	20.75
9	Korea	20.71
8	Japan	20.28
1	Austria	20.00
5	France	19.44
0	Australia	19.40
15	Spain	14.58
4	China	13.94
2	Brazil	7.17
13	Russia	7.09
10	Mexico	6.51
7	India	6.14

In [32]:

```
# plot a histogram of the countries

ax = data_plot.plot(x='Country',
                    kind='bar',
                    figsize=[12, 6],
                    width=0.9,
                    alpha=0.6,
                    color=color_list[6],
                    edgecolor='w',
                    grid=False,
                    ylim=[0, 30])

ax.set_xticks(range(len(data_plot)))
ax.set_xticklabels(data_plot.Country, rotation=80, rotation_mode='anchor', ha='right', fontproperties=ticks_font)
ax.yaxis.grid(True)
for label in ax.get_yticklabels():
    label.set_fontproperties(ticks_font)

ax.set_title('Autonomous Vehicles Countries by Readiness - Total Score ', fontproperties=title_font)
ax.set_xlabel('', fontproperties=label_font)
ax.set_ylabel('Total score', fontproperties=label_font)

one_patch = mpatches.Patch(color=color_list[6], label='Readiness - Total Score')
two_patch = mpatches.Patch(color='whitesmoke', label='KPMG Information 2018')
ax.legend(handles=[one_patch, two_patch])

plt.show()
```



Ranking Country by Policy and legislation Score

In [33]:

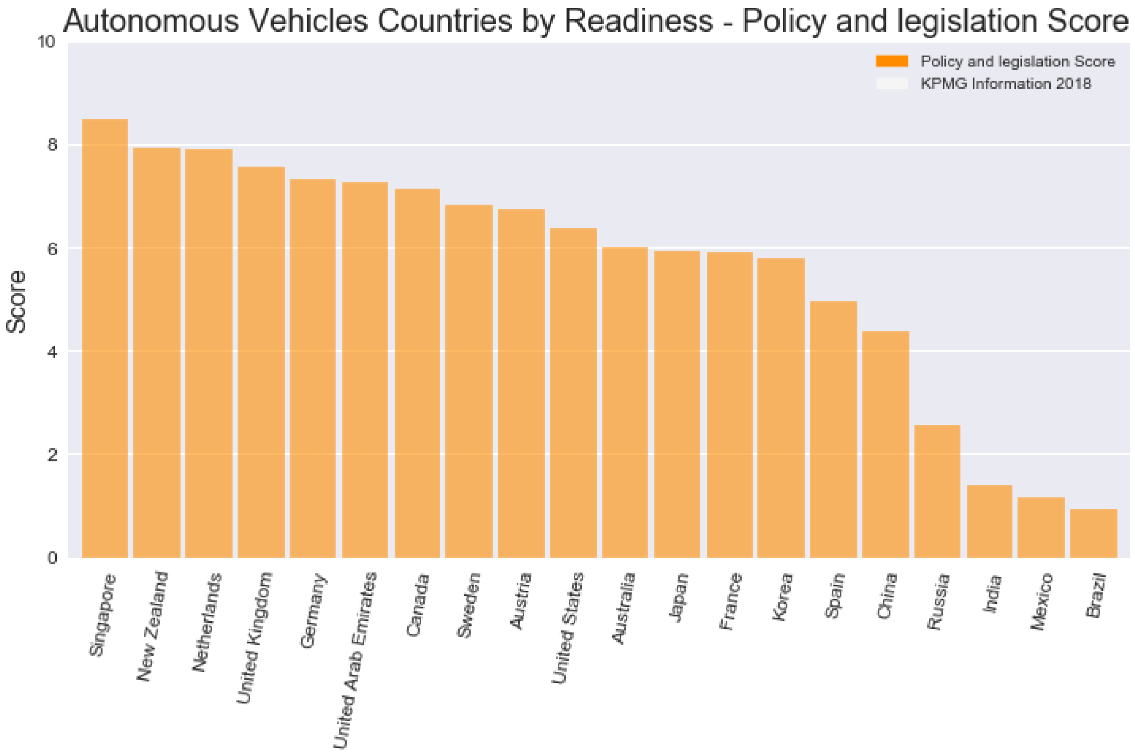
```
# Sort table by  rank
countdata = df_kpmg.sort_values(by=['Policy and legislation Score'],ascending=False)
# Filter first 20 lines
countdata = countdata[countdata['Policy and legislation Rank'] <= 20]

# make a copy with specified columns
data_plot = countdata[['Country', 'Policy and legislation Score']].copy()
data_plot
```

Out[33]:

	Country	Policy and legislation Score
14	Singapore	8.49
12	New Zealand	7.92
11	Netherlands	7.89
18	United Kingdom	7.55
6	Germany	7.33
17	United Arab Emirates	7.26
3	Canada	7.12
16	Sweden	6.83
1	Austria	6.73
19	United States	6.38
0	Australia	6.01
8	Japan	5.93
5	France	5.92
9	Korea	5.78
15	Spain	4.95
4	China	4.38
13	Russia	2.58
7	India	1.41
10	Mexico	1.16
2	Brazil	0.93

In [34]:

```
# plot a histogram of the countries

ax = data_plot.plot(x='Country',
                    kind='bar',
                    figsize=[12, 6],
                    width=0.9,
                    alpha=0.6,
                    color=color_list[8],
                    edgecolor='w',
                    grid=False,
                    ylim=[0, 10])

ax.set_xticks(range(len(data_plot)))
ax.set_xticklabels(data_plot.Country, rotation=80, rotation_mode='anchor', ha='right',
fontproperties=ticks_font)
ax.yaxis.grid(True)
for label in ax.get_yticklabels():
    label.set_fontproperties(ticks_font)

ax.set_title('Autonomous Vehicles Countries by Readiness - Policy and legislation Score
 ', fontproperties=title_font)
ax.set_xlabel('', fontproperties=label_font)
ax.set_ylabel('Score', fontproperties=label_font)

one_patch = mpatches.Patch(color=color_list[8], label='Policy and legislation Score')
two_patch = mpatches.Patch(color='whitesmoke', label='KPMG Information 2018')
ax.legend(handles=[one_patch, two_patch])

plt.show()
```



Ranking Country by Technology & innovation Score

In [35]:

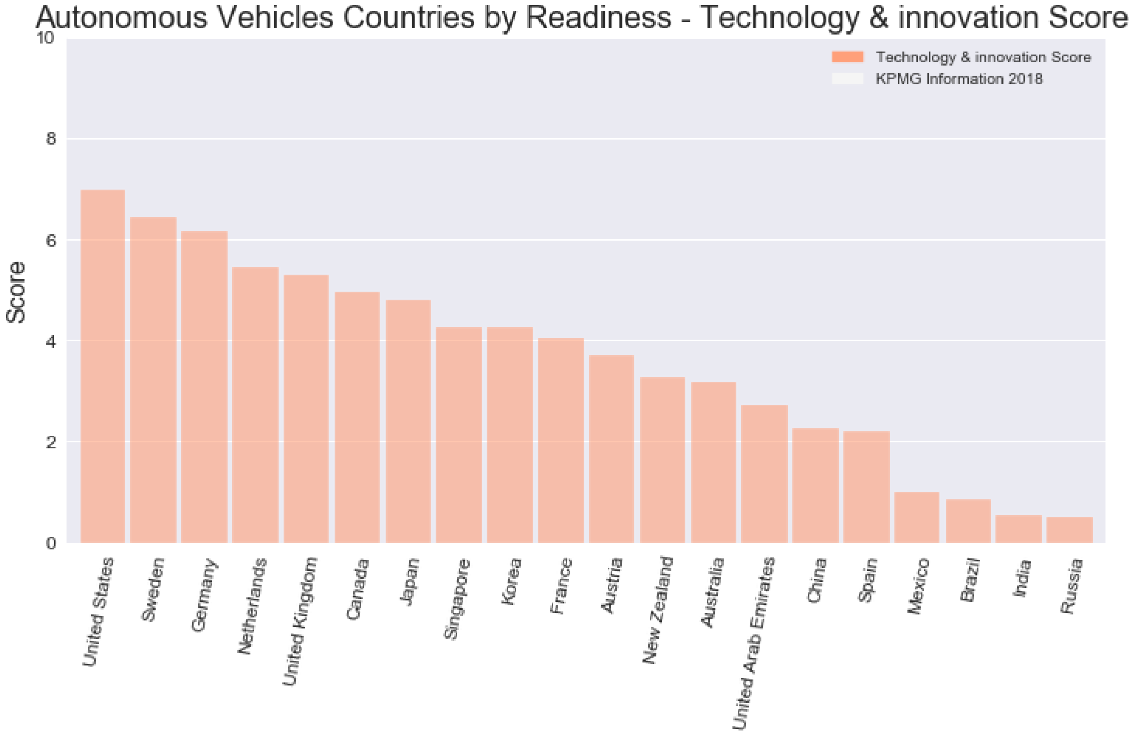
```
# Sort table by  rank
countdata = df_kpmg.sort_values(by=['Technology & innovation Score'],ascending=False)
# Filter first 20 lines
countdata = countdata[countdata['Technology & innovation Rank'] <= 20]

# make a copy with specified columns
data_plot = countdata[['Country', 'Technology & innovation Score']].copy()
data_plot
```

Out[35]:

	Country	Technology & innovation Score
19	United States	6.97
16	Sweden	6.44
6	Germany	6.15
11	Netherlands	5.46
18	United Kingdom	5.28
3	Canada	4.97
8	Japan	4.79
14	Singapore	4.26
9	Korea	4.24
5	France	4.03
1	Austria	3.69
12	New Zealand	3.26
0	Australia	3.18
17	United Arab Emirates	2.71
4	China	2.25
15	Spain	2.21
10	Mexico	1.01
2	Brazil	0.86
7	India	0.54
13	Russia	0.52

In [36]:

```
# plot a histogram of the countries

ax = data_plot.plot(x='Country',
                    kind='bar',
                    figsize=[12, 6],
                    width=0.9,
                    alpha=0.6,
                    color=color_list[2],
                    edgecolor='w',
                    grid=False,
                    ylim=[0, 10])

ax.set_xticks(range(len(data_plot)))
ax.set_xticklabels(data_plot.Country, rotation=80, rotation_mode='anchor', ha='right',
fontproperties=ticks_font)
ax.yaxis.grid(True)
for label in ax.get_yticklabels():
    label.set_fontproperties(ticks_font)

ax.set_title('Autonomous Vehicles Countries by Readiness - Technology & innovation Scor
e ', fontproperties=title_font)
ax.set_xlabel('', fontproperties=label_font)
ax.set_ylabel('Score', fontproperties=label_font)

one_patch = mpatches.Patch(color=color_list[2], label='Technology & innovation Score')
two_patch = mpatches.Patch(color='whitesmoke', label='KPMG Information 2018')
ax.legend(handles=[one_patch, two_patch])

plt.show()
```



Ranking Country by Infrastructure Score

In [37]:

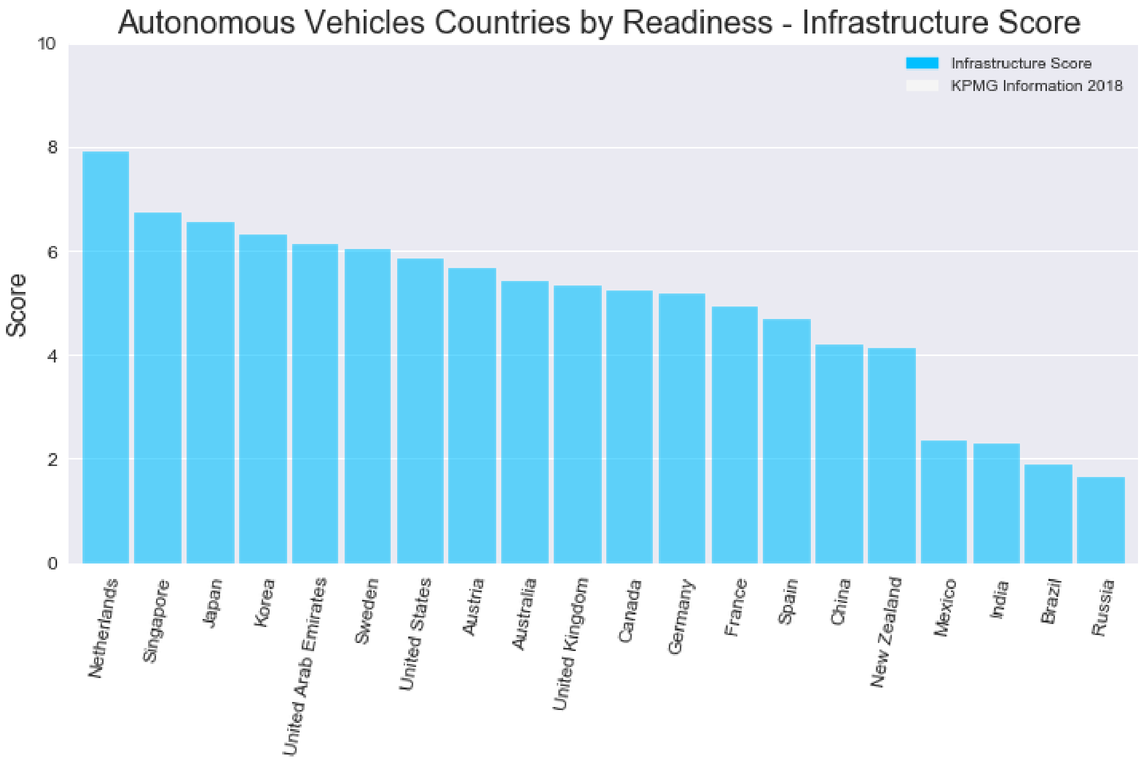
```
# Sort table by  rank
countdata = df_kpmg.sort_values(by=['Infrastructure Score'],ascending=False)
# Filter first 20 lines
countdata = countdata[countdata['Infrastructure Rank'] <= 20]

# make a copy with specified columns
data_plot = countdata[['Country', 'Infrastructure Score']].copy()
data_plot
```

Out[37]:

	Country	Infrastructure Score
11	Netherlands	7.89
14	Singapore	6.72
8	Japan	6.55
9	Korea	6.32
17	United Arab Emirates	6.12
16	Sweden	6.04
19	United States	5.84
1	Austria	5.66
0	Australia	5.43
18	United Kingdom	5.31
3	Canada	5.22
6	Germany	5.17
5	France	4.94
15	Spain	4.69
4	China	4.18
12	New Zealand	4.14
10	Mexico	2.34
7	India	2.28
2	Brazil	1.89
13	Russia	1.64

In [38]:

```
# plot a histogram of the countries
ax = data_plot.plot(x='Country',
                    kind='bar',
                    figsize=[12, 6],
                    width=0.9,
                    alpha=0.6,
                    color=color_list[4],
                    edgecolor='w',
                    grid=False,
                    ylim=[0, 10])

ax.set_xticks(range(len(data_plot)))
ax.set_xticklabels(data_plot.Country, rotation=80, rotation_mode='anchor', ha='right',
fontproperties=ticks_font)
ax.yaxis.grid(True)
for label in ax.get_yticklabels():
    label.set_fontproperties(ticks_font)

ax.set_title('Autonomous Vehicles Countries by Readiness - Infrastructure Score ', font
properties=title_font)
ax.set_xlabel('', fontproperties=label_font)
ax.set_ylabel('Score', fontproperties=label_font)

one_patch = mpatches.Patch(color=color_list[4], label='Infrastructure Score')
two_patch = mpatches.Patch(color='whitesmoke', label='KPMG Information 2018')
ax.legend(handles=[one_patch, two_patch])

plt.show()
```



Ranking Country by Consumer acceptance Score

In [39]:

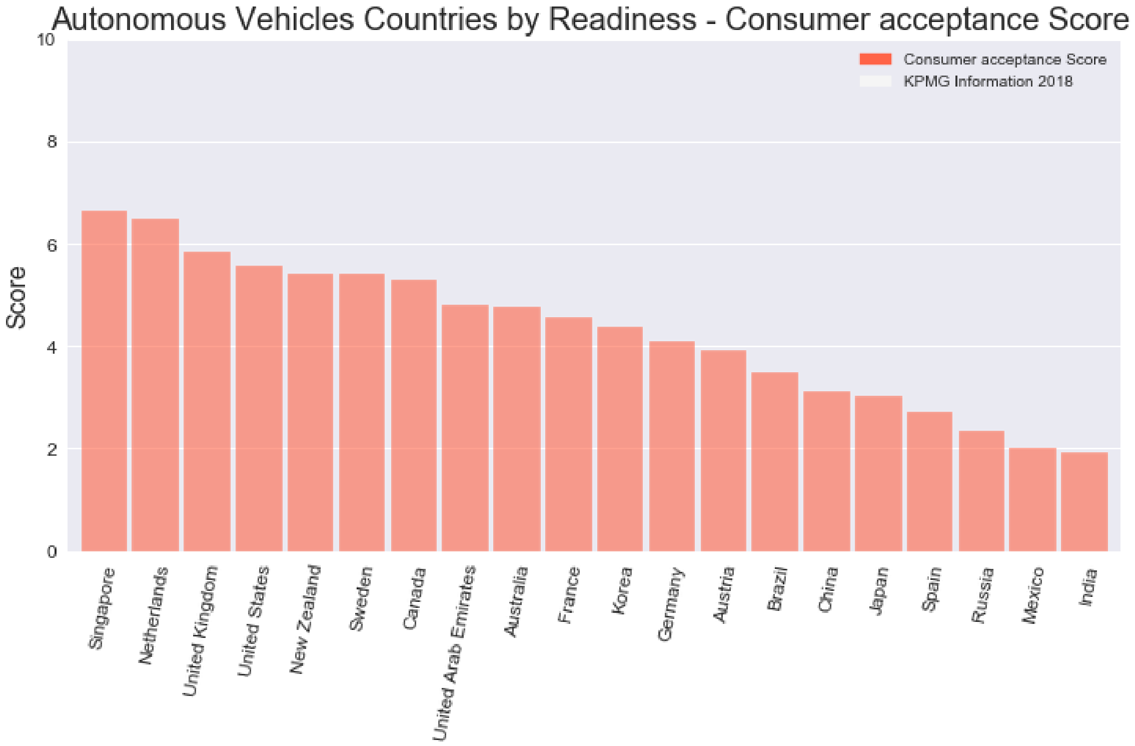
```
# Sort table by  rank
countdata = df_kpmg.sort_values(by=['Consumer acceptance Score'],ascending=False)
# Filter first 20 lines
countdata = countdata[countdata['Consumer acceptance Rank'] <= 20]

# make a copy with specified columns
data_plot = countdata[['Country', 'Consumer acceptance Score']].copy()
data_plot
```

Out[39]:

	Country	Consumer acceptance Score
14	Singapore	6.63
11	Netherlands	6.49
18	United Kingdom	5.84
19	United States	5.56
12	New Zealand	5.43
16	Sweden	5.41
3	Canada	5.30
17	United Arab Emirates	4.79
0	Australia	4.78
5	France	4.55
9	Korea	4.38
6	Germany	4.09
1	Austria	3.91
2	Brazil	3.49
4	China	3.13
8	Japan	3.01
15	Spain	2.72
13	Russia	2.35
10	Mexico	2.00
7	India	1.91

In [40]:

```
# plot a histogram of the countries
ax = data_plot.plot(x='Country',
                    kind='bar',
                    figsize=[12, 6],
                    width=0.9,
                    alpha=0.6,
                    color=color_list[1],
                    edgecolor='w',
                    grid=False,
                    ylim=[0, 10])

ax.set_xticks(range(len(data_plot)))
ax.set_xticklabels(data_plot.Country, rotation=80, rotation_mode='anchor', ha='right',
fontproperties=ticks_font)
ax.yaxis.grid(True)
for label in ax.get_yticklabels():
    label.set_fontproperties(ticks_font)

ax.set_title('Autonomous Vehicles Countries by Readiness - Consumer acceptance Score ',
 fontproperties=title_font)
ax.set_xlabel('', fontproperties=label_font)
ax.set_ylabel('Score', fontproperties=label_font)

one_patch = mpatches.Patch(color=color_list[1], label='Consumer acceptance Score')
two_patch = mpatches.Patch(color='whitesmoke', label='KPMG Information 2018')
ax.legend(handles=[one_patch, two_patch])

plt.show()
```



Ranking Country by Research and development AV Hubs

In [41]:

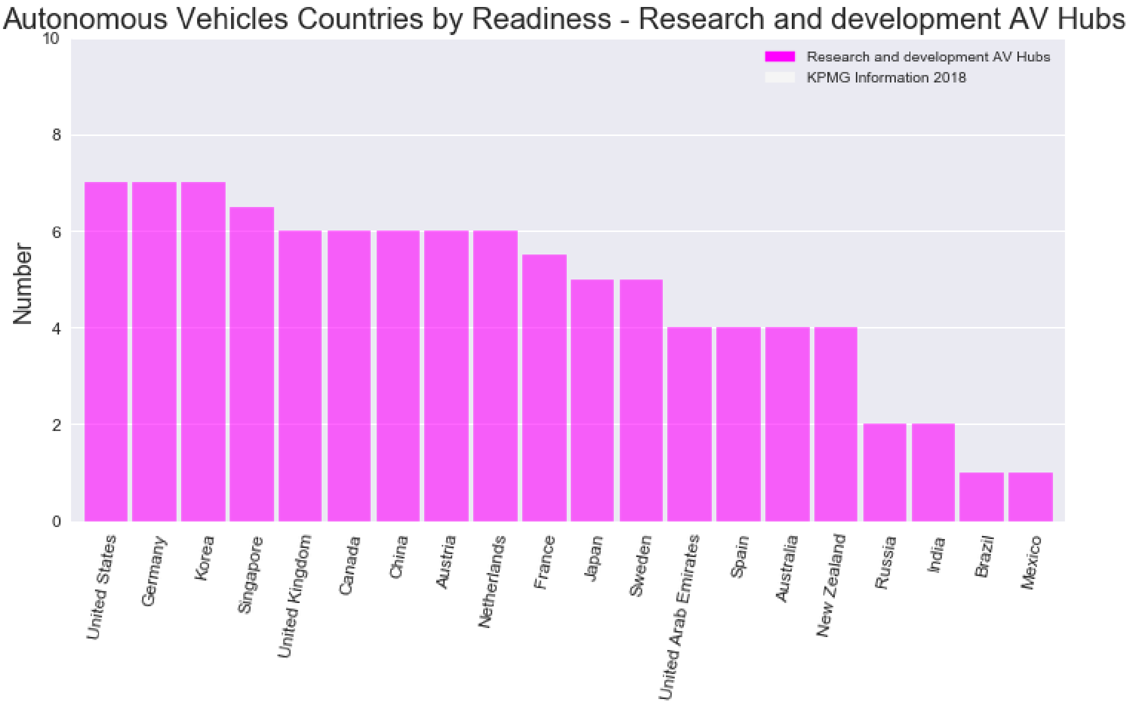
```
# Sort table by  rank
countdata = df_kpmg.sort_values(by=['Research and development AV Hubs'],ascending=False
)
# Filter first 20 lines
countdata = countdata[countdata['Research and development AV Hubs'] <= 20]

# make a copy with specified columns
data_plot = countdata[['Country', 'Research and development AV Hubs']].copy()
data_plot
```

Out[41]:

	Country	Research and development AV Hubs
19	United States	7.0
6	Germany	7.0
9	Korea	7.0
14	Singapore	6.5
18	United Kingdom	6.0
3	Canada	6.0
4	China	6.0
1	Austria	6.0
11	Netherlands	6.0
5	France	5.5
8	Japan	5.0
16	Sweden	5.0
17	United Arab Emirates	4.0
15	Spain	4.0
0	Australia	4.0
12	New Zealand	4.0
13	Russia	2.0
7	India	2.0
2	Brazil	1.0
10	Mexico	1.0

In [42]:

```
# plot a histogram of the countries
ax = data_plot.plot(x='Country',
                    kind='bar',
                    figsize=[12, 6],
                    width=0.9,
                    alpha=0.6,
                    color=color_list[3],
                    edgecolor='w',
                    grid=False,
                    ylim=[0, 10])

ax.set_xticks(range(len(data_plot)))
ax.set_xticklabels(data_plot.Country, rotation=80, rotation_mode='anchor', ha='right',
fontproperties=ticks_font)
ax.yaxis.grid(True)
for label in ax.get_yticklabels():
    label.set_fontproperties(ticks_font)

ax.set_title('Autonomous Vehicles Countries by Readiness - Research and development AV
 Hubs ', fontproperties=title_font)
ax.set_xlabel('', fontproperties=label_font)
ax.set_ylabel('Number', fontproperties=label_font)

one_patch = mpatches.Patch(color=color_list[3], label='Research and development AV Hub
s')
two_patch = mpatches.Patch(color='whitesmoke', label='KPMG Information 2018')
ax.legend(handles=[one_patch, two_patch])

plt.show()
```



Ranking Country by Connectivity Index for infrastructure¶

In [46]:

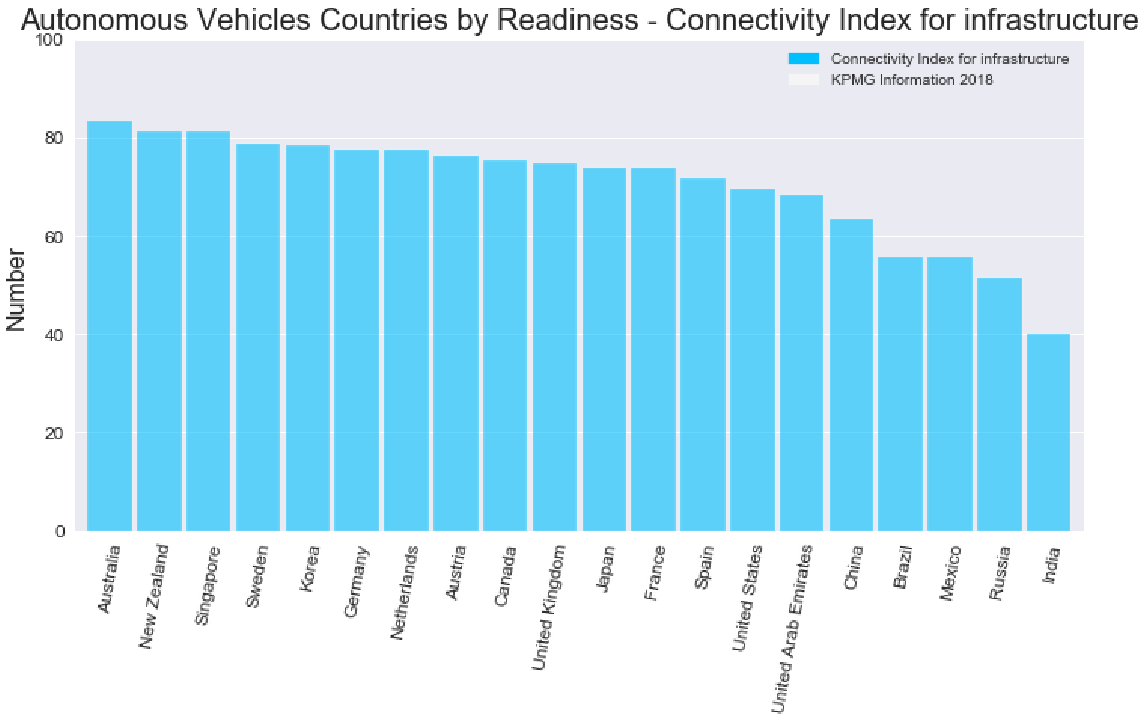
```
# Sort table by  rank
countdata = df_kpmg.sort_values(by=['Connectivity Index for infrastructure'],ascending=
False)
# Filter first 20 lines
#countdata = countdata[countdata['Connectivity Index for infrastructure'] <= 20]

# make a copy with specified columns
data_plot = countdata[['Country', 'Connectivity Index for infrastructure']].copy()
data_plot
```

Out[46]:

	Country	Connectivity Index for infrastructure
0	Australia	83.32
12	New Zealand	81.26
14	Singapore	81.14
16	Sweden	78.74
9	Korea	78.26
6	Germany	77.50
11	Netherlands	77.44
1	Austria	76.24
3	Canada	75.38
18	United Kingdom	74.76
8	Japan	73.84
5	France	73.66
15	Spain	71.53
19	United States	69.60
17	United Arab Emirates	68.14
4	China	63.51
2	Brazil	55.75
10	Mexico	55.67
13	Russia	51.56
7	India	39.99

In [49]:

```
# plot a histogram of the countries
ax = data_plot.plot(x='Country',
                    kind='bar',
                    figsize=[12, 6],
                    width=0.9,
                    alpha=0.6,
                    color=color_list[4],
                    edgecolor='w',
                    grid=False,
                    ylim=[0, 100])

ax.set_xticks(range(len(data_plot)))
ax.set_xticklabels(data_plot.Country, rotation=80, rotation_mode='anchor', ha='right',
fontproperties=ticks_font)
ax.yaxis.grid(True)
for label in ax.get_yticklabels():
    label.set_fontproperties(ticks_font)

ax.set_title('Autonomous Vehicles Countries by Readiness - Connectivity Index for infra
structure', fontproperties=title_font)
ax.set_xlabel('', fontproperties=label_font)
ax.set_ylabel('Number', fontproperties=label_font)

one_patch = mpatches.Patch(color=color_list[4], label='Connectivity Index for infrastru
cture')
two_patch = mpatches.Patch(color='whitesmoke', label='KPMG Information 2018')
ax.legend(handles=[one_patch, two_patch])

plt.show()
```



In [43]:

```
df_kpmg.head()
```

Out[43]:

	Overall rank	Country	Total score	Policy and legislation Rank	Policy and legislation Score	Technology & innovation Rank	Technology & innovation Score	Infrastru
0	14	Australia	19.40	11	6.01	13	3.18	9
1	12	Austria	20.00	9	6.73	11	3.69	8
2	17	Brazil	7.17	20	0.93	18	0.86	19
3	7	Canada	22.61	7	7.12	6	4.97	11
4	16	China	13.94	16	4.38	15	2.25	15

In [44]:

```
# Sort table by Overall rank
countdata = df_kpmg.sort_values(by=['Total score'],ascending=False)
# Filter first 20 lines
countdata = countdata[countdata['Overall rank'] <= 20]

# make a copy with specified columns
data_plot = countdata[['Country', 'Total score','Policy and legislation Score','Technol
ogy & innovation Score','Infrastructure Score','Consumer acceptance Score']].copy()
data_plot.head()
```

Out[44]:

	Country	Total score	Policy and legislation Score	Technology & innovation Score	Infrastructure Score	Consumer acceptance Score
11	Netherlands	27.73	7.89	5.46	7.89	6.49
14	Singapore	26.08	8.49	4.26	6.72	6.63
19	United States	24.75	6.38	6.97	5.84	5.56
16	Sweden	24.73	6.83	6.44	6.04	5.41
18	United Kingdom	23.99	7.55	5.28	5.31	5.84

In [45]:

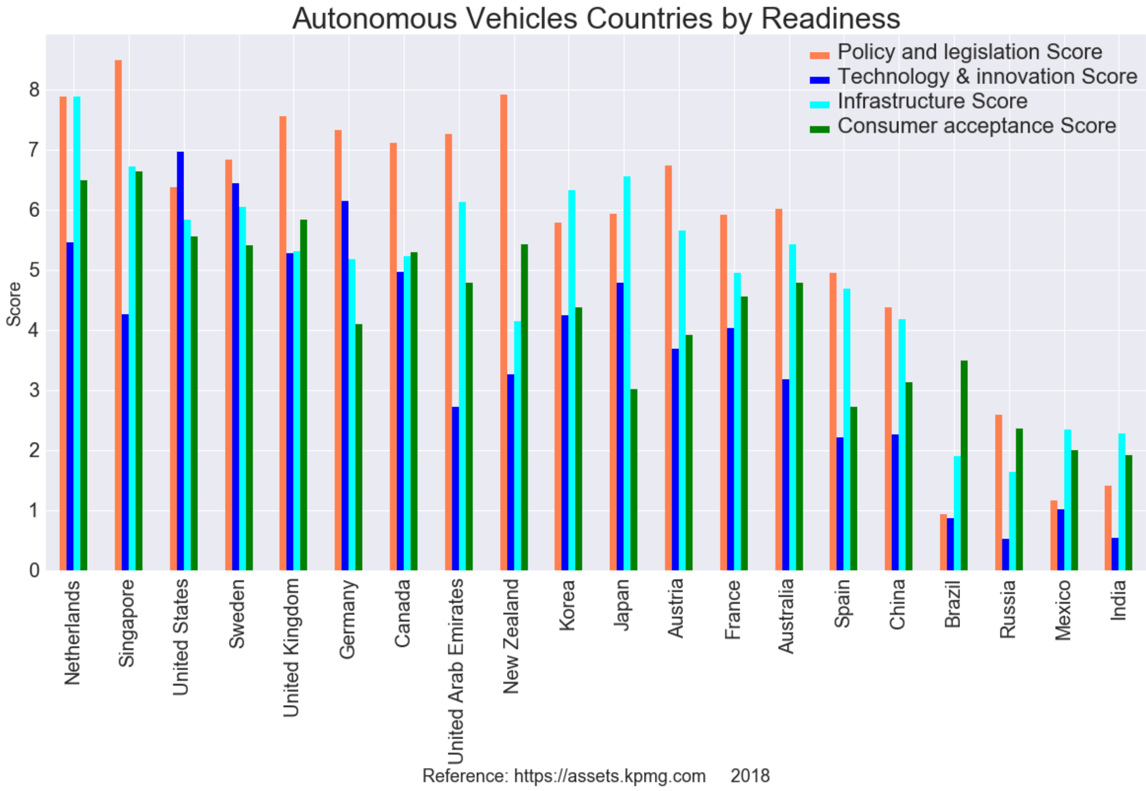
```
# delete column
data_plot_bk = data_plot.copy()
del data_plot['Total score']
data_plot
```

Out[45]:

	Country	Policy and legislation Score	Technology & innovation Score	Infrastructure Score	Consumer acceptance Score
11	Netherlands	7.89	5.46	7.89	6.49
14	Singapore	8.49	4.26	6.72	6.63
19	United States	6.38	6.97	5.84	5.56
16	Sweden	6.83	6.44	6.04	5.41
18	United Kingdom	7.55	5.28	5.31	5.84
6	Germany	7.33	6.15	5.17	4.09
3	Canada	7.12	4.97	5.22	5.30
17	United Arab Emirates	7.26	2.71	6.12	4.79
12	New Zealand	7.92	3.26	4.14	5.43
9	Korea	5.78	4.24	6.32	4.38
8	Japan	5.93	4.79	6.55	3.01
1	Austria	6.73	3.69	5.66	3.91
5	France	5.92	4.03	4.94	4.55
0	Australia	6.01	3.18	5.43	4.78
15	Spain	4.95	2.21	4.69	2.72
4	China	4.38	2.25	4.18	3.13
2	Brazil	0.93	0.86	1.89	3.49
13	Russia	2.58	0.52	1.64	2.35
10	Mexico	1.16	1.01	2.34	2.00
7	India	1.41	0.54	2.28	1.91

In [39]:

```
#Plot bar grahic with multi columns

ax = data_plot.plot( x= 'Country',
            kind='bar',
            figsize=(20,10),
            grid=True,
            color=('coral', 'blue','cyan','green'),   # One for each column
            fontsize=20)
plt.title('Autonomous Vehicles Countries by Readiness',fontsize=30)
plt.xlabel('Reference: https://assets.kpmg.com     2018',fontsize=18)
plt.ylabel('Score',fontsize=18)

# Setup size to legend
plt.setp(ax.get_legend().get_texts(), fontsize='22') # for legend text
plt.setp(ax.get_legend().get_title(), fontsize='22') # for legend title

plt.show()
```



In [40]:

```
# Join KPMG Statitics with world map

# Table WORLD - geography of World
# Table data_plot - Statitics KPMG
data_plot = data_plot_bk
map_kpmg = world.set_index('name').join(data_plot.set_index('Country'))
map_kpmg.head()
```

Out[40]:

	pop_est	continent	iso_a3	gdp_md_est	geometry	Tota scor
name						
Afghanistan	28400000.0	Asia	AFG	22270.0	POLYGON ((61.21081709172574 35.65007233330923,...	NaN
Angola	12799293.0	Africa	AGO	110300.0	(POLYGON ((16.32652835456705 -5.87747039146621...	NaN
Albania	3639453.0	Europe	ALB	21810.0	POLYGON ((20.59024743010491 41.85540416113361,...	NaN
United Arab Emirates	4798491.0	Asia	ARE	184300.0	POLYGON ((51.57951867046327 24.24549713795111,...	20.8
Argentina	40913584.0	South America	ARG	573900.0	(POLYGON ((-65.50000000000003 -55.199999999999...	NaN

In [41]:

```
# Sort table by Overall rank

map_kpmg1 = map_kpmg.sort_values(by=['Total score'],ascending=False)
```

In [42]:

```
map_kpmg1.head()
```

Out[42]:

	pop_est	continent	iso_a3	gdp_md_est	geometry	Tot sco
name						
Netherlands	16715999.0	Europe	NLD	672000.0	POLYGON ((6.074182570020923 53.51040334737814,...	27.7
United States	313973000.0	North America	USA	15094000.0	(POLYGON ((-155.54211 19.08348000000001, -155....	24.7
Sweden	9059651.0	Europe	SWE	344300.0	POLYGON ((22.18317345550193 65.72374054632017,...	24.7
United Kingdom	62262000.0	Europe	GBR	1977704.0	(POLYGON ((-5.661948614921897 54.5546031764838...	23.9
Germany	82329758.0	Europe	DEU	2918000.0	POLYGON ((9.921906365609232 54.98310415304803,...	22.7

```
In [43]:
```

```
map_kpmg2 = map_kpmg1.sort_values(by=['Total score'],ascending=False)
map_kpmg2.head()
```

```
Out[43]:
```

	pop_est	continent	iso_a3	gdp_md_est	geometry	Tot sco
name						
Netherlands	16715999.0	Europe	NLD	672000.0	POLYGON ((6.074182570020923 53.51040334737814,...	27.7
United States	313973000.0	North America	USA	15094000.0	(POLYGON ((-155.54211 19.08348000000001, -155....	24.7
Sweden	9059651.0	Europe	SWE	344300.0	POLYGON ((22.18317345550193 65.72374054632017,...	24.7
United Kingdom	62262000.0	Europe	GBR	1977704.0	(POLYGON ((-5.661948614921897 54.5546031764838...	23.9
Germany	82329758.0	Europe	DEU	2918000.0	POLYGON ((9.921906365609232 54.98310415304803,...	22.7

In [44]:

```
# Includind and Index
map_kpmg3 = map_kpmg2.reset_index()
map_kpmg3.head()
```

Out[44]:

	name	pop_est	continent	iso_a3	gdp_md_est	geometry	T sc
0	Netherlands	16715999.0	Europe	NLD	672000.0	POLYGON ((6.074182570020923 53.51040334737814,...	27
1	United States	313973000.0	North America	USA	15094000.0	(POLYGON ((-155.54211 19.08348000000001, -155....	24
2	Sweden	9059651.0	Europe	SWE	344300.0	POLYGON ((22.18317345550193 65.72374054632017,...	24
3	United Kingdom	62262000.0	Europe	GBR	1977704.0	(POLYGON ((-5.661948614921897 54.5546031764838...	23
4	Germany	82329758.0	Europe	DEU	2918000.0	POLYGON ((9.921906365609232 54.98310415304803,...	22

In [45]:

```
# change the geometry to point
points_map = map_kpmg1.copy()

# Only firts 20 countries
points_map = points_map.head(20)

# POINT Geometry
points_map.geometry = points_map['geometry'].centroid
#points.crs = poly.crs
points_map.head()
```

Out[45]:

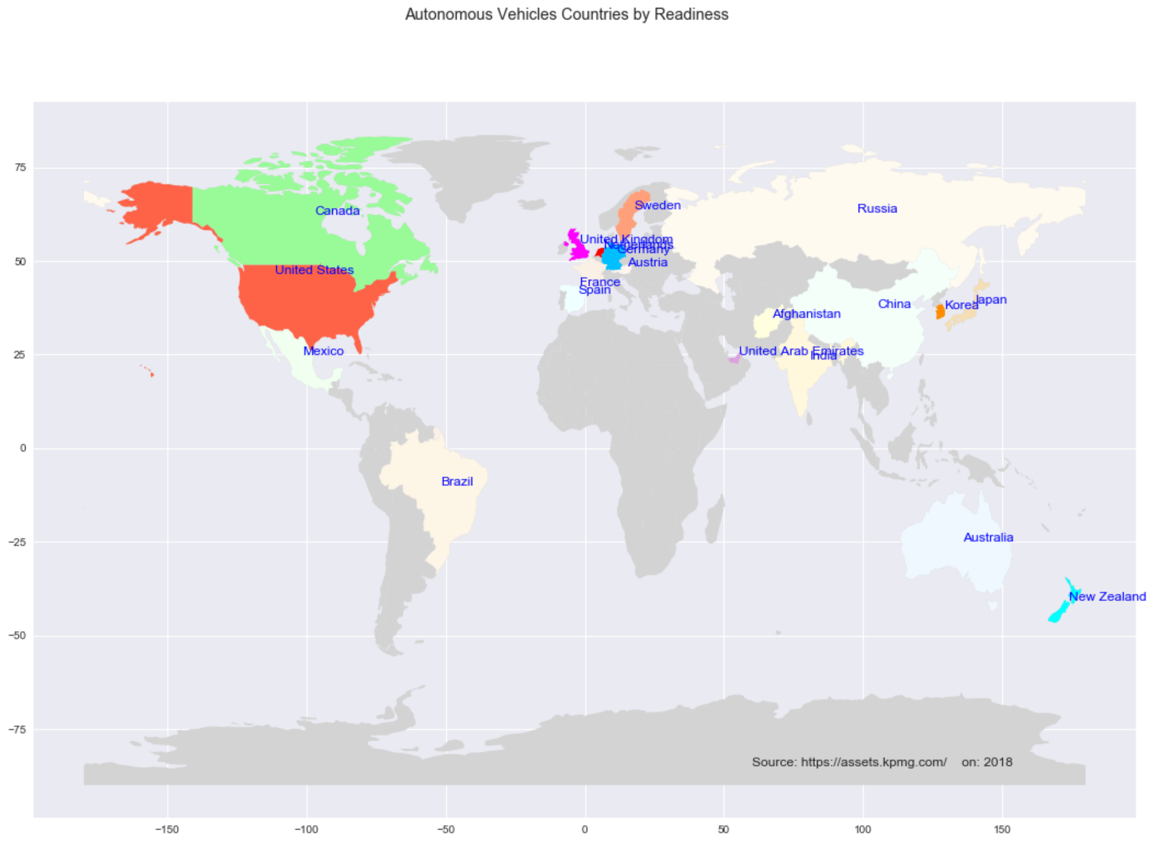
	pop_est	continent	iso_a3	gdp_md_est	geometry	Tota scor
name						
Netherlands	16715999.0	Europe	NLD	672000.0	POINT (5.512217100965401 52.2987003744418)	27.73
United States	313973000.0	North America	USA	15094000.0	POINT (-112.5994383773273 45.70562953540318)	24.75
Sweden	9059651.0	Europe	SWE	344300.0	POINT (16.59626584684802 62.81148496808033)	24.73
United Kingdom	62262000.0	Europe	GBR	1977704.0	POINT (-2.853135395180542 53.91477348053708)	23.99
Germany	82329758.0	Europe	DEU	2918000.0	POINT (10.28848509274289 51.13372269040782)	22.74

In [46]:

```
# Plot One MAP

# Defining two area to plot
f, ax1 = plt.subplots(ncols=1, sharex=False, sharey=False, figsize=(18,12))
ax1.set_axis_on()
f.suptitle('Autonomous Vehicles Countries by Readiness')

#-- First Graph -----
# PLOT State points
# Plot the states area
ax1 = map_kpmg3.plot(ax=ax1, facecolor='blue', alpha=1, linewidth=0, color = 'lightgrey') # cmap='YlOrRd')    cmap=plt.cm.rainbow
for i in np.arange(0, 20):  # Print 15 registres
    # 1 registo
    ax1 = map_kpmg3[map_kpmg3.index == i].plot(ax=ax1,  alpha=1, linewidth=1, color=color_list[i],legend = True)


# Plot the labels - Only the names to the countries
for x, y, label in zip(points_map.geometry.x, points_map.geometry.y, points_map.index):
    ax1.annotate(label, xy=(x, y), xytext=(3, 3), alpha=3, textcoords="offset points",color='blue')

ax1.text(60, -85, 'Source: https://assets.kpmg.com/    on: 2018')
```

Out[46]:

```
<matplotlib.text.Text at 0x2657ab7c4a8>
```



In [ ]: